# Oracle® Services Tools Bundle (STB)

User's Guide

Release 8.09 for Oracle Solaris

**E35226-16**

September 2015

ORACLE®

Oracle Services Tools Bundle (STB) User's Guide, Release 8.09 for Oracle Solaris

E35226-16

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

# Contents

**Preface** ..... v

Audience ..... v
Documentation Accessibility ..... v
Conventions ..... v

## 1 Oracle Services Tools Bundle Overview

1.1 Oracle STB Components ..... 1-1
1.2 Download and Install Oracle Services Tools Bundle ..... 1-2
1.2.1 Non-Global Zone Installation ..... 1-4
1.2.2 Using the `STB_INST_PATH` Environment Variable ..... 1-4
1.2.3 Command Line Interface Options of the Oracle STB Installer ..... 1-4
1.2.4 Extracting Packages ..... 1-5
1.2.5 IPS Actions, Triggered by STB Installation ..... 1-5

## 2 Oracle Explorer Data Collector

2.1 Standard Oracle Explorer Data Collector Installation ..... 2-2
2.1.1 Non-standard Installations ..... 2-2
2.2 Oracle Explorer Data Collector Configuration ..... 2-3
2.2.1 Location of RDA ..... 2-3
2.2.2 Explorer Software Directory ..... 2-3
2.2.3 Explorer Configuration Directory ..... 2-4
2.2.4 Explorer Temporary Directory ..... 2-4
2.2.5 Explorer Output Directory ..... 2-4
2.2.6 Explorer Target Directory and Collection Result File ..... 2-5
2.2.7 Configure cron to Run Explorer Regularly ..... 2-5
2.3 Using Oracle Explorer Data Collector ..... 2-5
2.3.1 Run Explorer Proactively ..... 2-5
2.3.2 Run Explorer Reactively ..... 2-5
2.3.3 Run Explorer for Different Modules/Groups ..... 2-5
2.4 Submitting Explorer Collection Result Files ..... 2-6
2.4.1 Using the Diagnostic Assistant (DA) ..... 2-6
2.4.2 Using HTTP/HTTPS ..... 2-7
2.4.3 Using Secure File Transport (SFT) ..... 2-7
2.4.4 Using FTP ..... 2-7
2.5 Explorer Security Guidelines ..... 2-7

## 3 Oracle SNEEP


3.1 SNEEP Downloads and Product Support ..... 3-2
3.2 Accessing the Serial Number ..... 3-2
3.2.1 EEPROM on x86/x64 Platforms ..... 3-2
3.3 Using SNEEP ..... 3-3
3.3.1 Storing Serial Number Data in EEPROM ..... 3-3
3.3.2 Storing Other Information ..... 3-4
3.3.3 Removing or Erasing a SNEEP Tag and Value ..... 3-4
3.3.4 Sharing SNEEP over NFS ..... 3-4
3.3.5 Using SNEEP with Solaris 10 Zones ..... 3-5
3.3.6 Using SNEEP on Fujitsu PRIMEPOWER Hardware ..... 3-6


## 4 Troubleshooting STB and STB Components


4.1 Troubleshooting STB Installation ..... 4-1
4.1.1 STB Error Messages ..... 4-1
4.1.2 STB Check Sum Error ..... 4-1
4.1.3 The Installation Log ..... 4-2
4.1.4 Possible Issues with IPS Installation ..... 4-3
4.1.5 Tracing an STB Installation ..... 4-3
4.1.6 Package Removal Error in STB Log File *(Solaris 11 or Later, Only)* ..... 4-4
4.2 Troubleshooting Explorer Collections ..... 4-4
4.2.1 Tracing an Explorer Collection ..... 4-4
4.2.2 Finding Explorer Modules in RDA ..... 4-5
4.2.3 Explorer Error Messages ..... 4-5
4.2.4 ACT Truncated Output ..... 4-6
4.2.5 Running Oracle Explorer Data Collector in the Background ..... 4-6
4.2.6 Explorer Hangs on Zone pkg.ksh (Solaris Only) ..... 4-6
4.3 Troubleshooting SNEEP ..... 4-6
4.3.1 SNEEP Performance ..... 4-6
4.3.2 EEPROM `nvramrc` Formatting with Veritas VxVM ..... 4-7
4.3.3 SNEEP Returns Serial Number `unknown` ..... 4-8
4.3.4 Preserving SNEEP Settings After EEPROM Restore or Replace ..... 4-8
4.3.5 Preserving SNEEP Settings After OBP Firmware Update ..... 4-9
4.3.6 Unable to Set `hostname` and `hostid` Tags ..... 4-9
4.3.7 Overriding the Size Limit of the EEPROM `nvramrc` Variable ..... 4-9
4.3.8 SNEEP Reports a `Bad String` ..... 4-9


## A Oracle Explorer Data Collector Third-Party License Agreement


A.1 cURL 7.22.0 ..... A-1
A.2 OpenSSL 1.0.1p ..... A-1


## Index

# Preface

The Oracle Services Tools Bundle (STB) is a self-extracting installer bundle that supports all Solaris standard operating systems and architectures, enabling customers to get the most from their Oracle Premier Support plans.

This document describes how to download and install Oracle STB. The following information is provided:

- Oracle Services Tools Bundle Overview
- Oracle Explorer Data Collector
- Oracle SNEEP
- Troubleshooting STB and STB Components

## Audience

This document is intended for systems administrators who are responsible for maintaining hardware with the Oracle Solaris platform.

## Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc`.

### Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` or visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` if you are hearing impaired.

## Conventions

The following text conventions are used in this document:

| Convention | Meaning |
|---|---|
| **boldface** | Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary. |
| *italic* | Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values. |

| Convention | Meaning |
| --- | --- |
| `monospace` | Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter. |

# 1

# Oracle Services Tools Bundle Overview

The Oracle Services Tools Bundle (STB) is a self-extracting installer bundle that supports all Solaris standard operating systems and architectures, enabling customers to get the most from their Oracle Premier Support plans.

With Oracle STB, you can realize the following key benefits:

- Expedites problem diagnosis and resolution.
- Proactive prevention.
- Easy installation of all the latest tools.

This chapter covers the following topics:


- Oracle STB Components
- Download and Install Oracle Services Tools Bundle


## 1.1 Oracle STB Components

Each release of Oracle STB includes the latest versions of the component parts that make up STB. Each component is specifically designed to analyze or gather information for efficient problem resolution. A component can group more than one product. For example, the Oracle Explorer Data Collector groups the Explorer and the Remote Diagnostic Agent (RDA) products.

- Oracle Explorer Data Collector

  The Oracle Explorer Data Collector is a collection of shell scripts and a few binary executables that gathers information and creates a detailed snapshot of a system's configuration and state. See Chapter 2, "Oracle Explorer Data Collector," for details.

- Oracle Remote Diagnostic Agent (RDA)

  RDA is a command-line diagnostic platform that is executed by an engine written in the Perl programming language. It provides a unified package of support diagnostics tools and preventive solutions. The data captured provides a comprehensive picture of the your environment, which aids in problem diagnosis.

  See the *RDA Getting Started Guide* (Doc ID 314422.1) in My Oracle Support for details:

  `https://support.oracle.com`

- Oracle Serial Number in EEPROM (SNEEP)

Oracle Serial Number in EEPROM (SNEEP) provides a software-accessible Chassis Serial Number (CSN) for all Oracle Solaris hardware platforms. See Chapter 3, "Oracle SNEEP," for details.

- Other STB components

  Depending on the specific architecture and version of the target system, other components for STB may be included, such as:

  – Oracle Autonomous Crashdump Tool (ACT)

    Oracle ACT is an extension to `mdb(1)` which creates a readable summary of a Solaris kernel crash dump. It will automatically process in the background any new kernel crash dump it finds on reboot after a system panic.

    See *Oracle Autonomous Crashdump Tool (ACT)* (Doc ID 1438006.1) in My Oracle Support for more information:

    `https://support.oracle.com`

  – Oracle Service Tags.

## 1.2 Download and Install Oracle Services Tools Bundle

Oracle Services Tools Bundle is available as a single installer bundle from My Oracle Support.

`https://support.oracle.com`

---

**Note:** A login is required for accessing content on My Oracle Support.

---

To install the Oracle Services Tools Bundle:

1. Download and unzip the Oracle Services Tools Bundle from Doc ID 1153444.1. The latest available version of Oracle STB provides for:

   - Oracle Solaris: 8, 9, 10, 11 Express, and 11

     – Includes both 32-bit and 64-bit

     ---

     **Note:** To determine if your version of Oracle Solaris is 32 or 64 bit, run:

     ```
     example% isainfo -v
     ```

     Output will look like:

     ```
     64-bit sparcv9 applications
     32-bit sparc applications
     ```

     To determine the kernel modules, run:

     ```
     example% isainfo -vk
     ```

     Output will look like:

     ```
     64-bit sparcv9 kernel modules
     ```

     ---

   - SPARC and x86 infrastructure

2. Log in as `root`.
3. Make sure the install script is executable:

   ```
   chmod +x install_stb.sh
   ```

4. Run the following command:

   ```
   ./install_stb.sh -verbose
   ```

You will be asked to choose between **I**(install), **X**(extract), or **E**(exit). Choose the **I(install)** option. The installer will transparently install:

- On Oracle Solaris 11 Express, 11, or higher:
  - RDA and Explorer via an image package system (IPS) package with the following command as a leading part of the Fault Management Resource Identifier (FMRI):

    ```
    pkg://solaris/support/explorer
    ```

    The software locations are `/usr/lib/explorer` for the Explorer part and `/usr/lib/rda` for the RDA part. Three symbolic links are created:

    * ***From*** `/usr/sbin/explorer` ***to*** `/usr/lib/explorer/bin/explorer`
    * ***From*** `/usr/bin/rda` ***to*** `/usr/lib/rda/rda.sh`
    * ***From*** `/usr/bin/sdci` ***to*** `/usr/lib/rda/sdci.sh`

  - SNEEP via an IPS package with the following command as leading part of the FMRI.

    ```
    pkg://solaris/support/sneep
    ```

    The software location is the `/usr/lib/sneep` directory. A symbolic link is created from `/usr/sbin/sneep` to `/usr/lib/sneep/bin/sneep`.

  - ACT via the IPS package with the following command as leading part of the FMRI.

    ```
    pkg://solaris/support/act
    ```

    The software location is the `/usr/lib/act` directory.

- On Oracle Solaris 8, 9, and 10:

  **Note:** When installing via IPS, the software is always installed in the default directories. When installing via SVR4, you can modify the location of the installation directory by installing the packages manually via `pkgadd`.

  - Explorer and RDA via the packages `SUNWexplo`, `SUNWexplu`, and `SUNWrda`. The default software locations are the `/opt/SUNWexplo`, `/etc/opt/SUNWexplo`, and `/usr/lib/rda` directories.
  - SNEEP via the package `SUNWsneep`. The standard software location is the `/opt/SUNWsneep` directory.
  - ACT is shipped via the `CTEact` package. The standard software location is the `/opt/CTEact` directory.

Depending on your software installation preferences, you have several options for installing the Oracle Explorer Data Collector, including:

- Non-Global Zone Installation
- Using the STB_INST_PATH Environment Variable
- Command Line Interface Options of the Oracle STB Installer
- Extracting Packages
- IPS Actions, Triggered by STB Installation

### 1.2.1 Non-Global Zone Installation

Beginning with Oracle Solaris 10, non-global zones are supported. However, Oracle Explorer, ACT, and RDA can only be installed in the global zone. SNEEP can be installed in a non-global zone.

### 1.2.2 Using the STB_INST_PATH Environment Variable

To install the software packages, the `install_stb.sh` installation script will extract the software archive. By default, it uses the `/tmp/STBinstall` directory. You can specify an alternative extraction directory by setting the `STB_INST_PATH` environment variable, in which case the `$STB_INST_PATH/tmp/STBinstall` directory will be used as the extraction directory.

---

**Note:** The extracted software archive will be deleted after the execution of the `install_stb.sh` script. You can indicate that you want to keep the extraction archive by setting the environment variable `STB_INST_KEEP` to **1**.

---

### 1.2.3 Command Line Interface Options of the Oracle STB Installer

The following command options are available to the Oracle STB installation script:

```
install_stb.sh [-verbose] [-trace]
install_stb.sh -version
install_stb.sh -V
install_stb.sh -help
install_stb.sh -runmode manual [-force] [-trace] [-verbose]
install_stb.sh -runmode auto [-force] [-trace] [-verbose]
install_stb.sh -runmode auto -ext [architecture,version] [-trace] [-verbose]
```

Where:

- `-verbose` will make the STB installer more verbose. This option is highly recommended.
- `-version` displays the `README.txt` file. The `README.txt` file contains release and build information on the STB components.
- `-V` displays release and build of STB.
- `-help` displays the online help man pages for STB.
- `-runmode manual` runs STB in manual mode (default)
- `-runmode auto` installs packages in auto/non-interactive mode

- `-runmode auto -ext` extracts packages for the current or the specified architecture and operating system version combination (for example, Solaris 9 on SPARC or Solaris 11 on x86)
- `-trace` will run the STB installer, which is a shell script, in trace mode. This option is mainly used for debugging purposes.

**Notes:**

- When installing via the `install_stb.sh` script, all relevant products for this platform will be considered for installation/upgrade.
- When you specify the `-force` option on Solaris 8, 9, and 10, STB will always uninstall the currently installed SVR4 packages and install the bundled versions. Without the `-force` option, the SVR4 packages will only be uninstalled if the SVR4 package in the extraction archive is more recent than the installed SVR4 package.

### 1.2.4 Extracting Packages

To extract individual packages from the Oracle STB software bundle:

1. Log in as root.
2. Make sure the install script is executable:

   ```
   chmod +x install_stb.sh
   ```

3. Run `./install_stb.sh -verbose`
4. Choose **X**(extract), and the packages specific for the current architecture and OS version will be extracted.

   By default the `/var/tmp/stb/extract` directory will be used but you can use the `STB_EXT_PATH` environment variable to change this destination. In this case, the software will be extracted to the `$STB_EXT_PATH/stb/extract` directory.

   - For Solaris 11 or later, a file-based IPS repository will be extracted, which contains the IPS packages for RDA/Explorer, SNEEP, and ACT. On top of this, one package stream containing a Service Tag package is extracted.
   - For Solaris 8, 9, and 10 systems:

     SVR4 package streams will be extracted. For example, you will find `/var/tmp/stb/extract/Explorer/SUNW.<version>.pkg` (RDA is considered as part of the Explorer distribution).

When using the command-line options, it is possible to specify an alternative extraction platform. For example, you can extract the SVR4 package streams for a SPARC Solaris 10 system on a Solaris 11 system.

### 1.2.5 IPS Actions, Triggered by STB Installation

For systems running Solaris 11 or later, most installation tasks are done through the Image Packaging System (IPS). The steps taken for IPS are:

- Publish the file-based IPS repository, extracted from the `install_stb.sh` script. Run:

  ```
  pkg set-publisher -g <ips repository> solaris
  ```

- For software that is in IPS format, the installer will launch:

  ```
  ips install <software>
  ```

- When all the software is installed, the file-based repository is unpublished with:

  ```
  pkg set-publisher -G <ips-repository> solaris
  ```

> **Note:** The individual IPS actions can take time, which might give the impression that the STB installer on Solaris 11 is slow or even hanging.

# 2

# Oracle Explorer Data Collector

This chapter provides information for installing, configuring, and running the Oracle Explorer Data Collector. The following topics are discussed:


- Standard Oracle Explorer Data Collector Installation
  - Non-standard Installations
- Oracle Explorer Data Collector Configuration
  - Location of RDA
  - Explorer Software Directory
  - Explorer Configuration Directory
  - Explorer Temporary Directory
  - Explorer Output Directory
  - Explorer Target Directory and Collection Result File
- Using Oracle Explorer Data Collector
  - Run Explorer Proactively
  - Run Explorer Reactively
  - Run Explorer for Different Modules/Groups
- Submitting Explorer Collection Result Files
  - Using the Diagnostic Assistant (DA)
  - Using Secure File Transport (SFT)
  - Using HTTP/HTTPS
  - Using FTP
- Explorer Security Guidelines


### Additional Oracle Explorer Data Collector Resources

You can find additional information about the Oracle Explorer Data Collector in the My Oracle Support knowledgebase, including:

- *Oracle Explorer Data Collector: Modules, Module Groups, Module Aliases Summary* (Doc ID 1536532.1)
- *Oracle Explorer Data Collector Implementation Best Practice* (Doc ID 1006990.1)
- *How to Upload Data to Oracle Such as Explorer and Core Files* (Doc ID 1020199.1)

Log in to My Oracle Support at:

https://support.oracle.com

A set of man pages are included with the Oracle Explorer Data Collector. To access the man pages in Solaris versions prior to Solaris 11, you can add the location of the Explorer man pages to the MANPATH environment variable, or you can indicate the location of the Explorer man pages explicitly in the man command.

**Solaris 8, 9, and 10**

The man pages are located in the /opt/SUNWexplo/man directory for Solaris 8, 9, and 10.

There are three man pages for explorer in sections 1m, 4, and 5.

For example, to access the Explorer man page of section 1m on Solaris 10, run:

```
man -M /opt/SUNWexplo/man -s 1m explorer
```

Or, if the MANPATH environment variable is set, the command would be:

```
man -s 1m explorer
```

**Solaris 11**

For Solaris 11 and later, man pages are already on the default man path.

The man pages are located in the /usr/lib/explorer/man directory for Solaris 11 and later.

To access the Explorer man pages using the default man path, run:

```
man explorer
```

To access the Explorer man page of section 4 on Solaris 11, run:

```
man -s 4 explorer
```

## 2.1 Standard Oracle Explorer Data Collector Installation

The preferred method for installing Oracle Explorer Data Collector is via the Oracle Services Tools Bundle (STB), because Oracle Explorer relies on the presence of the other software packages on the system.

Packages, which are installed via STB, depend on the architecture and operating system on which you install.

---

**Note:** The Oracle Explorer Data Collector is supported on Oracle Solaris Releases 8, 9, 10, 11 Express, and 11.

---

See Download and Install Oracle Services Tools Bundle for instructions to download and install Oracle STB.

### 2.1.1 Non-standard Installations

It is possible to install the Oracle Explorer Data Collector without installing other STB components, but since the Oracle Explorer Data Collector delegates its collection work to the Oracle Remove Diagnostic Agent (RDA), it is necessary to install RDA together with Explorer to have the full functionality of Explorer.

See Download and Install Oracle Services Tools Bundle for instructions to download and install Oracle STB.

To install Explorer on Solaris 11 and Solaris 11 Express systems:

1. Extract the file-based IPS repository.
2. Install the support/explorer package from the IPS repository.

To install Explorer on Solaris 8, 9, and 10 systems:

1. Extract the SVR4 package streams for the Explorer component (`SUNWexplo`, `SUNWexplu`, `SUNWrda`).
2. Install these SVR4 package streams with the `pkgadd` command.

## 2.2 Oracle Explorer Data Collector Configuration

After you have installed Explorer and RDA, you will need to configure Explorer. This section provides configuration instructions for the following Explorer components:

- Location of RDA
- Explorer Software Directory
- Explorer Configuration Directory
- Explorer Temporary Directory
- Explorer Output Directory
- Explorer Target Directory and Collection Result File
- Configure cron to Run Explorer Regularly

### 2.2.1 Location of RDA

Explorer needs RDA for all of its collections and will look for RDA on the standard locations. Explorer will check for the `RDA_HOME` entry in the `rdainput.txt` file. When no entry exists, the following locations are scanned for a valid RDA distribution:

- The `../rda` directory.
- If the `support/explorer` IPS package is installed, then the `/usr/lib/rda` directory.
- If the `SUNWrda` package is installed, then the `BASEDIR(SUNWrda)/rda` directory.
- The `/usr/lib/rda` directory.

---

**Note:** The RDA installation includes the Diagnostic Assistant, which is the recommended for submitting Explorer collection result files to Oracle Support. For more information, see *Diagnostic Assistant: General Information* (Doc ID 201804.1) in My Oracle Support:

`https://support.oracle.com`

---

### 2.2.2 Explorer Software Directory

When Explorer is installed via standard packaging operation, the following locations are used:

- `/usr/lib/explorer` on Solaris 11 and Solaris 11 Express (IPS installation).
- `/opt/SUNWexplo` on Solaris 8, 9, and 10 (SVR4 installation).

It is possible to move the files to a different location; however, you will need to set the `EXP_SOFT` environment variable.

### 2.2.3 Explorer Configuration Directory

Explorer needs a configuration directory to put its configuration files. You can set the `EXP_CONFIG` environment variable to indicate the location of the configuration directory. When this environment variable is not set, the following locations are used:

- `/etc/explorer` on Solaris 11, Solaris 11 Express, or higher (IPS installation).
- `/etc/opt/SUNWexplo` on Solaris 8, 9, and 10 (SVR4 installation).

The following configuration information can be present:

- An **Oracle Wallet** can be present to store Explorer passwords. The management of the Wallet is delegated to the Diagnostic Assistant (DA). The wallet is stored in the `da` subdirectory of the Explorer configuration directory.
- An Oracle Wallet can be present to store Explorer passwords. We call this wallet the **Explorer wallet**. The management of the Explorer wallet is delegated to the Diagnostic Assistant (DA). The wallet is stored in the `da` subdirectory of the Explorer configuration directory. Since it is possible to have multiple Explorer configuration directories, you can have multiple Explorer wallets.
- The **module input files** contain information for the configuration of a module (for example, information on the hostname of the system controller with credential information).

  The location and name of the input files can be altered by command line arguments of Explorer and by environment variables. Check the `man` pages of Explorer for details.
- The **defaults file** contains general configuration information for Explorer. Its name is "explorer" and the file is located in the "default" subdirectory of the Explorer configuration directory.

### 2.2.4 Explorer Temporary Directory

Explorer needs a temporary, but secure directory. You can set the `EXP_TMP` environment variable to indicate its location. When `EXP_TMP` is not set, the following locations are used:

- `/var/explorer/tmp` on Solaris 11 and Solaris 11 Express (IPS installation).
- `/opt/SUNWexplo/tmp` on Solaris 8, 9, and 10 (SVR4 installation).

> **Note:** If you choose to use a different location for the temporary directory, it must have `0700` protection and be owned by `root`.

### 2.2.5 Explorer Output Directory

To store its collection result file, Explorer will create a subdirectory in the output directory of Explorer. The following locations are used for the output directory:

- `/var/explorer/output` on Solaris 11 or later
- `/opt/SUNWexplo/output` on Solaris 8, 9, and 10.

It is possible to specify an alternative location of the output directory in the defaults file or by using a command line argument of Explorer.

### 2.2.6 Explorer Target Directory and Collection Result File

The collection result of Explorer is stored in a subdirectory of the Explorer output directory, which contains the host ID, the host name, and a time stamp. For example:

```
/var/explorer/output/explorer.843bf20a.xx-ttttt-1-2013.11.05.13.50
```

The last step in the Explorer collection is to package this target directory in the collection result file in compressed tape archive (tar) format.

A typical name this compressed package is:

```
explorer.843bf20a.xx-tttt-1-2013.11.05.13.50.tar.gz
```

### 2.2.7 Configure cron to Run Explorer Regularly

Most users run Explorer as part of best practices for system administration. You can create a `cron` entry for Explorer to run during off-peak hours. For example:

```
# cron
```

This command sets the Explorer job to:

- Run every day at 2 a.m.
- Run Explorer on every module.
- Send the Explorer collection result file to Oracle Support using the Diagnostic Assistant, which is defined in the defaults file in the Explorer Configuration Directory.

## 2.3 Using Oracle Explorer Data Collector

This section provides details for how to run Oracle Explorer Data Collector.

The Oracle Explorer Data Collector is designed to run at a regularly scheduled time (proactive) or manually, on demand (reactive).

### 2.3.1 Run Explorer Proactively

In a proactive environment, you can run Explorer at regular intervals typically via a `cron` job.

### 2.3.2 Run Explorer Reactively

You can run Explorer manually, on demand (reactive) for any reason at any time for any modules and groups. Typically, though, you would run Explorer reactively as part of the troubleshooting associated with a particular Service Request. For example, to run Explorer on all modules, use the Diagnostic Assistant to upload the collection result file to Oracle Support, and associate it with a Service Request:

```
# explorer -w all -T DA -SR <Service Request number>
```

### 2.3.3 Run Explorer for Different Modules/Groups

Explorer can be run for the following modules/groups:

- `explorer -w all`

  Runs all modules.

- `explorer -w all,interactive`

If the modules tagged to the group all require user interaction, the user is prompted for input.

- `explorer -w default`

  Runs modules tagged to default

- `explorer -w default,interactive`

  Runs modules tagged to default group. If the modules tagged to the group default require user interaction, the user is prompted for input.

- `explorer -w extended`

  Runs modules tagged to mandatory or extended group.

- `explorer -w <module name>`

  Runs `<module name>` and modules tagged to the mandatory group.

- `explorer -w default,<module name>`

  Modules tagged under the group `default` and `module_name` are executed. Even if the modules tagged under the group `default` require user interaction, the user is NOT prompted for input. If the `module_name` requires user input (the corresponding configuration file `*input.txt` is not populated with relevant information), the user is prompted for input; if the `module_name` does not require user input, it is executed if hardware compatibility is met.

## 2.4 Submitting Explorer Collection Result Files

After the Oracle Explorer Data Collector has created a collection result file, it can be sent to a different server. You can use the defaults file in the Explorer Configuration Directory to define your preferred method of transport.

You can transfer Explorer collection results:

- Immediately after the collection via the `-P` command-line option.
- At any time when an Explorer collection exists via the `-N` command-line option.

You can send the Explorer result directly to Oracle, which requires an internet connection, or you can send it to a different server. The actual transfers to Oracle are in that case done from this other server.

The following methods are supported:

- Using the Diagnostic Assistant (DA) - Recommended when Explorer collections are sent directly from this machine to Oracle. The method is available in Explorer 8.02 or later.
- Using HTTP/HTTPS
- Using Secure File Transport (SFT)
- Using FTP

### 2.4.1 Using the Diagnostic Assistant (DA)

The Diagnostic Assistant (DA) is included as part of the RDA installation of STB. This recommended feature takes advantage of Oracle's common upload destination (`transport.oracle.com`).

To set up DA as the default transport method, you must create or update the Explorer defaults file with the `explorer -g` command. The DA transport is proposed as the

default transport since Explorer 8.02. Alternatively you can use the `explorer -T DA` command line option to use the DA transport for this run of Explorer.

To upload Explorer collection result files to Oracle, the My Oracle Support (MOS) username and password must be known. To store both username and password and to provide other information to upload files to MOS, go to the `da` subdirectory of the RDA distribution and run the following command:

```
./da.sh setupmos
```

Enter at least the MOS username and password. After this configuration step, Explorer can upload collection results to Oracle by using DA.

For more information, see *Diagnostic Assistant: General Information* (Doc ID 201804.1) in My Oracle Support:

`https://support.oracle.com`

### 2.4.2 Using HTTP/HTTPS

You can use http/https to transfer files from the Explorer machine to a different server. Setting it up is analogous to setting DA up.

This transport method is mainly used for non-authenticated HTTPS or FTPS methods to a different server.

### 2.4.3 Using Secure File Transport (SFT)

Oracle Secure File Transport (SFT) supports data transfer between customer environments and Oracle. The SFT tool is used to periodically deliver Explorer data collector files for proactive reporting and for sending Explorer, core, log, or other files for support services diagnostics. SFT is designed to support customer network environments in which Explorer clients do not have Internet access and to provide a central point to manage Explorer telemetry.

In other words, the SFT server is a *hub*. Typically, it will accept Explorer collection results from other servers and send them to Oracle.

For details, see the *Oracle® Secure File Transport User's Guide*:

`http://docs.oracle.com/cd/E49687_01/index.htm`

### 2.4.4 Using FTP

It is possible to transfer files via FTP. It is up to the customer to asses the risks in using this transport method. It is no longer possible to use FTP to send files directly to Oracle.

## 2.5 Explorer Security Guidelines

1. Assess the privileges for the Explorer distribution. Only the user, who installs the software, needs write access to the software directory structure. All other users, including users who run Explorer collections, must only have read access to the Explorer distribution with execute privileges on the directories and the executables.
2. Make sure that the Explorer secure temporary directory is owned by `root` with permission set to **700** as protection.

3. Make sure that the Explorer configuration directory structure is owned by `root` and only accessible by `root`. The input files and credential store are ignored when `group` or others can access them. The same applies to the input files specified by command line argument or environment variable. The configuration directory and the input files should not be located on a Network Files System (NFS) drive.
4. When you need credentials for your Explorer collections, make sure to create the credential store (Explorer wallet) and to store your credentials in it. Ensure that you remove all passwords from the input files.
5. Explorer creates the collection results in the Explorer output directory so that they are only accessible by `root`. Place the output directory on a location where those restrictions can be enforced.
6. When connecting to external devices over an open network, use a preferred secure protocol such as `ssh` or `ftps` instead of `telnet` and `ftp`. Avoid well-known credentials for external devices.
7. Assess the transfer of the Explorer collection results to other boxes. Ensure that confidentiality and the integrity of the data is preserved. When transferring the collection results - especially when sending over the Internet, use a protocol where the data are encrypted with strong ciphers and transfer to servers where the identity can be verified.

# 3

# Oracle SNEEP

Oracle Serial Number in EEPROM (SNEEP) provides a software-accessible Chassis Serial Number (CSN) for all Oracle Solaris hardware platforms.

SNEEP uses the system EEPROM for persistent storage of the Chassis Serial Number and other important user-defined data such as asset information, contract ID, or the serial numbers of attached storage devices.

The presence of the software-accessible serial number and other service-related information can significantly simplify activities related to system service and asset management.

Without SNEEP, only a subset of the Solaris-based hardware platforms have a mechanism to maintain a software-accessible serial number. Among those platforms, there is a wide variety of mechanisms for this, making consistent access to this information difficult.

SNEEP provides one simple and consistent interface to the management of this information on all Solaris hosts, domains, and zones. SNEEP can also reference and maintain the serial number in the configuration files for the Oracle Explorer Data Collector.

The following topics are discussed:


- SNEEP Downloads and Product Support
- Accessing the Serial Number
- Using SNEEP


### Additional Oracle SNEEP Resources

You can find additional information about Oracle SNEEP in the My Oracle Support knowledgebase, including:

- *Frequently Asked Questions about SNEEP* (Doc ID 1294842.1)

Log in to My Oracle Support at:

`https://support.oracle.com`

To access the man page of Oracle SNEEP, you can add the location of the SNEEP man page to the `MANPATH` environment variable, or you can indicate the location of the SNEEP man page explicitly in the `man` command. The man page is located in:

- For Solaris 8, 9, and 10, the path is:

  `/opt/SUNWsneep/man`

- For Solaris 11 and higher, the path is:

```
/usr/lib/sneep/man
```

The man page for Oracle SNEEP is in section **1m**. For example, to access the SNEEP man page on Solaris 11, run:

```
man -M /usr/lib/sneep/man -s 1m sneep
```

Or, if the `MANPATH` environment variable is set, the command would be:

```
man -s 1m sneep
```

## 3.1 SNEEP Downloads and Product Support

SNEEP is included in the Oracle Services Tools Bundle (STB). See Download and Install Oracle Services Tools Bundle for instructions to download and install STB.

For support to SNEEP, you can submit a service request through My Oracle Support:

`https://support.oracle.com`

## 3.2 Accessing the Serial Number

If your hardware includes the serial number in its EEPROM, then you can view it with the following command:

```
sneep -t hostname,serial,model
```

If your hardware does not already store the serial number in EEPROM, then you can use SNEEP to include it. You will need to first obtain the serial number and manually enter it into EEPROM.

On most Sun hardware, the serial number is always located on a physical tag somewhere on the machine. The Sun System Handbook tells exactly where to find it for all Sun machines. The handbook can be accessed at My Oracle Support (click the **Knowledge** tab and then select Sun **System Handbook** in the left-hand navigation):

`https://support.oracle.com`

The data sheet (usually yellow or orange), which was delivered with your machine, shows the serial number and system configuration.

**WARNING: The decimal number reported in the SPARC boot banner as the serial number when the machine is powered on is NOT the true serial number - it is the decimal form of the "host id". The host id is ordinarily seen in hexadecimal as the output of the Solaris `hostid` command.**

### 3.2.1 EEPROM on x86/x64 Platforms

Many x86/x64 platforms provide hardware support for the serial number which is accessible using `smbios` or `ipmitool`. If possible, SNEEP will get the serial from these data sources and will use it to reload the simulated EEPROM automatically.

While x86 PC platforms do not have an EEPROM, Solaris x86 has an EEPROM emulation which is managed in a file by the `eeprom` command:

```
/boot/solaris/bootenv.rc
```

By default, the simulated EEPROM does not have an `nvramrc` variable, but `eeprom` is willing to create one for SNEEP.

Because Solaris has this simulated EEPROM, you can use SNEEP for the same tags and in the same way that you use it on other systems. Just be aware that the values are not really stored in EEPROM and are "lost" if the file system is badly damaged or recreated.

If you have been using Oracle Explorer Data Collector, then the SNEEP backup file will be saved as part of any Explorer files which you may have archived. Restoring that file (`etc/default/SUNWSNEEP`) will immediately allow SNEEP to retrieve the values and reload the simulated EEPROM.

## 3.3 Using SNEEP

You can use SNEEP to retrieve serial number information or you can use SNEEP to add such information to EEPROM. You do not need root access to retrieve the information; however, you may need root or superuser access to add the information to EEPROM, depending on the hardware.

To retrieve serial number information from EEPROM using SNEEP, run:

```
sneep
```

Follow the procedures below to use SNEEP to store serial number and other information in EEPROM.

---

**Note:** SNEEP does not validate the serial number, manufacturing date and location, etc., information. However, it does make an effort to ensure that it is approximately the right length and does not have any obviously invalid characters in it.

---

### 3.3.1 Storing Serial Number Data in EEPROM

SNEEP uses the Open Boot Prom (OBP) `nvramrc` variable to store a specially formatted `print` command for each tag/value pair. These commands may look odd because the OBP programming language is FORTH and the `<print>` command in FORTH is `<.">`. For example:

```
." ChassisSerialNumber ABCD1234 " cr
```

The `eeprom` command is used very carefully to store and retrieve these strings, taking care to preserve any other `nvramrc` contents.

It is possible to use the OBP `oem-banner` variable to store the serial number or other information. This variable has the advantage that it is not erased when the OBP defaults are restored, but it has very limited size. The limited size makes it difficult to manage multiple data items.

In addition, when the `oem-banner` variable is set, the system administrator loses the ability to get the Ethernet MAC address from the default OBP banner. This can be a very real disadvantage when working with a new system, or trying to debug a network-boot problem.

### 3.3.2 Storing Other Information

Another data item useful in many Data Center environments is the location of each machine on the Data Center floor, often specified as a grid coordinate like "B6". This would be stored and then retrieved using something like this:

```
$ sneep -t GRIDLOCATION -s B6
$ sneep -t GRIDLOCATION
B6
```

When opening a service case on external storage attached to a system, Oracle requires the serial number of the storage, not the system. Assume that there are two StorEdge 3510 arrays on a machine using controller 6, targets 2 and 4. For example:

```
$ sneep -t SE3510_c6t2_PSN -s 085581
$ sneep -t SE3510_c6t4_PSN -s 002D11
```

Storing other tags depends on what else is in the `nvramrc`, the length of the tags and values, and on the specific hardware platform. Different platforms use different EEPROM chips. Smaller EEPROM (300 bytes) could store 10 SNEEP tag/value pairs if the average size of each was approximately 30 bytes. Newer machines tend to have a larger EEPROM, around 4000 bytes.

If there are other items in the `nvramrc` variable (for example, `devalias` device alias entries), then these reduce the amount of space available for SNEEP tags.

SNEEP limits your total `nvramrc` usage to a conservative amount appropriate for your hardware platform, unless more than that is already in use. If that is the case, then you can use no more than is already in use, unless you override this safety limit with the `-F` (force) option.

### 3.3.3 Removing or Erasing a SNEEP Tag and Value

A data item is removed by setting it a null value using a pair of quotes with nothing between them:

```
$ sneep -t GRIDLOCATION -s B6
$ sneep -t GRIDLOCATION
B6
$ sneep -t GRIDLOCATION -s ""
$ sneep -t GRIDLOCATION
unknown
$
```

### 3.3.4 Sharing SNEEP over NFS

It is possible to share SNEEP over NFS; however, SNEEP is properly installed by means of a `pkgadd`, but technically, it does not have to be installed in order to be used. For example, you can run it directly from the subdirectory in the expanded package if you wish to do so. So yes, SNEEP can be used from an NFS mount point.

However, if it is not installed, the system startup links are not installed, and you will not have the boot-time automatic data integrity checking, automatic recovery, and automatic update of the optional Oracle Explorer plug-in.

In some environments, having these features can significantly reduce the time and effort required for maintenance of the serial number. It depends on whether your platform provide hardware support for the serial number, and/or how common it is to overwrite the Explorer configuration, or to reset the EEPROM.

**Note:** Flash archives often include the configuration files from the flash source machine, and installing these files through the flash process has the same effect as overwriting them.

In the absence of hardware-supported serial number, the flashed machine will report the serial of the flash source machine until corrected manually with the `sneep -s` command.

## 3.3.5 Using SNEEP with Solaris 10 Zones

In the global zone, SNEEP acts just as it would on any other Solaris domain. There are a few differences with non-global zones:

- Since SNEEP R2.5_1.79, the `pkgadd` command will not automatically add the SNEEP package to non-global zones. It is recommended that the SUNWSNEEP package be added to each non-global zone individually to preserve "administrative consistency" so that the same procedures can be used to retrieve the serial number and other tags without regard for the type of domain or zone. The end-user should not need to know this kind of detail.
- Non-global zones cannot update the EEPROM.

### SPARC

Non-global zones can read the EEPROM data. As long as the data stored for the zone is the same as the data already in EEPROM, then the EEPROM will be used. If the data to be stored for the zone is different from the data in EEPROM, or if the SNEEP tag used in a non-global zone is absent from the EEPROM, the data will be stored in the SNEEP backup file in the non-global zone. In this way the system eeprom is used as much as possible, but it can be supplemented or overridden as needed by the SNEEP backup file in a non-global zone.

### x86/x64

x86/x64 platforms have no true eeprom and non-global zones lack the openprom driver for the missing hardware. Non-global zones cannot read data from the global zone's simulated EEPROM. On these platforms, the SNEEP backup file has to function as the EEPROM. All SNEEP data for the non-global zone is stored in the SNEEP backup file. No SNEEP data will be available unless it has been explicitly set in the non-global zone using "sneep [-t tag] -s <setting>". For administrative consistency (so that you can use SNEEP the same way in every domain and zone) we recommend that when you install SNEEP in each non-global zone, you set the serial number and any other tags useful in your environment.

### All Platforms

- SNEEP tags are not persistent across OS reload.

  A side effect of the eeprom handling in non-global zones is that SNEEP tags in these zones which are not also present in the hardware EEPROM are not persistent in cases where the Operating System is replaced or the zone is recreated.

  Upgrades implemented by patching will not affect SNEEP tags because the SNEEP backup file is not disturbed.

- File systems inherited from the global zone.

  If you are installing or configuring SNEEP in a sparse non-global zone which inherits file systems from the global zone, you may see messages indicating that the installer was unable to create files in the inherited (read-only) file system.

In the normal case, this will not affect SNEEP as long as SNEEP is installed in the global zone using the corresponding place in the file system of the global zone. We recommend that you install SNEEP into the global zone first, so that the installer can see that the SNEEP files are already in place.

- **Example 1:** default case: `/usr` is inherited, `/opt` is not SNEEP installs properly in its own `/opt`. SNEEP cannot create a link to itself in `/usr/sbin`. However, a `/usr/sbin/SNEEP` link created in the global zone can be used in the non-global zone.
- **Example 2:** `/usr` is inherited, `/opt` is also inherited SNEEP cannot be installed in the read-only `/opt`. SNEEP cannot create a link to itself in `/usr/sbin`. However, `/opt/SUNWSNEEP` and a `/usr/sbin/SNEEP` link created in the global zone can be used in the non-global zone just as though the installation had been successful.

### 3.3.6 Using SNEEP on Fujitsu PRIMEPOWER Hardware

Generally, the Fujitsu PRIMEPOWER platform requires you to have special privilege to read its serial number. However, once the serial number has been set into EEPROM by SNEEP, you can get it back from EEPROM without special permissions.

Nevertheless, you will need root permission to get the actual serial number:

```
fserial
```

Once you have the serial number, then use SNEEP to enter it into EEPROM:

```
sneep
```

If the serial has been retrieved previously by SNEEP using `fserial` as the root user, and saved into the EEPROM, then it will be reported correctly. Otherwise, you might be seeing a value (from an Explorer file, for example) which is different from the true value which would be reported to the root user.

# 4

# Troubleshooting STB and STB Components

This chapter describes common troubleshooting steps for Oracle Services Tools Bundle (STB) and associated components. The following troubleshooting resolution steps are provided:

- Troubleshooting STB Installation
- Troubleshooting Explorer Collections
- Troubleshooting SNEEP

## 4.1 Troubleshooting STB Installation

Refer to the sections below for information on troubleshooting your Oracle Services Tools Bundle installation:

- STB Error Messages
- STB Check Sum Error
- The Installation Log
- Possible Issues with IPS Installation
- Tracing an STB Installation
- Package Removal Error in STB Log File (Solaris 11 or Later, Only)

### 4.1.1 STB Error Messages

It is possible that an error occurs when you are running Explorer. Most STB install error messages have the format `STB-nnnnn`. You can get more information on this as follows:

```
explorer -X STB-nnnnn
```

For example:

- **Error message:** `STB-02004: Cannot publish the IPS repository`
- **Cause:** The IPS repository in the installation area cannot be published.
- **Action:** Check that the origins of all active IPS repositories on this box are reachable.

### 4.1.2 STB Check Sum Error

**Error message:** `STB-00010: Incorrect payload check sum`

**Cause:** This error message indicates that the STB payload, which was extracted from the installation script, does not have the correct check sum. If the check sums do not match, then it is most likely that the STB bundle has become corrupted. Possible causes for such a file corruption include:

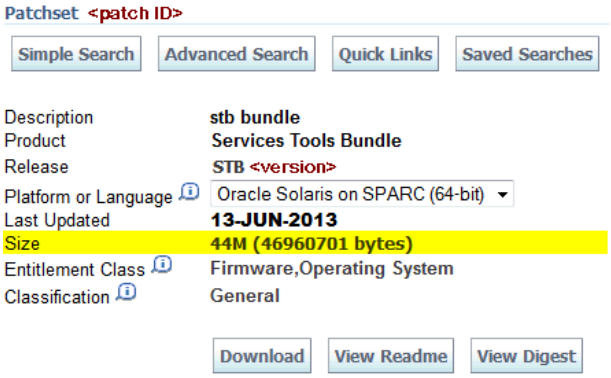
- If the STB bundle was downloaded onto a non-Solaris machine (such as a PC), and then copied to the Solaris machine, then the copying process from one system to the other may have encountered an error. Such an error may contribute to the file corruption.
- If the STB bundle was downloaded via FTP to a Solaris system, then it could become corrupted if you did not use `bin` (binary) and instead transferred the file as ASCII text.
- If the STB bundle was downloaded to a PC, then the download settings of the PC may have automatically unzipped the bundle, which could cause the file to become corrupted.

**Action:** In most cases, you can download the STB bundle again and retry the installation. If possible, download the STB bundle from Oracle directly to the Solaris system that you want to install STB.

1. Select a platform to download from Knowledge Article 1153444.1. You can access this article in My Oracle Support:

   `https://support.oracle.com`

2. From the Patchset README, note the size of the STB bundle:




   In this example for the SPARC 64-bit platform, the size is **44M (46960701 bytes)**.

3. Click the **Download** button to download the file.
4. In the directory where you downloaded the STB bundle, run the `ls -l` command. The output for this command would look like this:

```
# ls -l

-rw-r--r--   1 root     root    46960701 Jun 15 12:12 <patchID>_SOLARIS64.zip
```

The file size from the `ls -l` command output should match the size you note in the patchset README.

### 4.1.3 The Installation Log

When errors occur during the installation, you can review the installation log, which keeps track of all attempted installations. The log is located in `/var/log` with a file name `install_stb-v<STB-VERSION>.log` (for example, `/var/log/install_`

stb-v7.0.log). This log can provide you with more information on an issue. For every install that you do, the installer writes log messages to this file. Make sure that you only inspect the relevant (probably last) part of the log file.

> **Note:** It is also possible to trace the STB installation. See Tracing an STB Installation.

### 4.1.4 Possible Issues with IPS Installation

Other IPS installation error examples include:

- STB-02004: Cannot publish the IPS repository
- STB-02023: IPS installation of support/sneep failed
- STB-02023: IPS installation of support/act failed
- STB-02013: Cannot unpublish the IPS repository

Typically you will find lines like this in the log file:

```
Installing support/explorer ...

   Errors were encountered while attempting to retrieve
   package or file data for the requested operation.
   Details follow:

   http protocol error: code: 404 reason: Not Found
   URL: 'http://host1.mycompany.com/solaris11/release/
   solaris/manifest/0/support%2Fexplorer@6.10%2C5.11-12.4.17%3A20120417T105756Z'.
   (happened 4 times)

   STB-02023: IPS installation of support/explorer failed
```

If you get such an error message while installing STB and see lines like the above ones in the log file, the most likely cause is that there is an active IPS repository on the system, which can no longer be reached. To list the active IPS repositories, run:

```
# pkg publisher
```

Output should look like:

```
PUBLISHER             TYPE     STATUS   URI
solaris               origin   online   file:///var/tmp/stb/extract/ipsrepo/
```

In this case, we see a file-based IPS repository point to /var/tmp/stb/extract/ipsrepo. If someone has removed this file, the IPS actions, triggered by the STB installation, will all fail.

A similar case can happen when an HTTP-based repository can no longer be reached.

The remedy in this cases is to "deactivate" the non-reachable IPS repository by:

```
pkg set-publisher -G <ips repository> <publisher>
```

### 4.1.5 Tracing an STB Installation

You might find the STB installation slow or might even think the installation is hanging.

If you thing this is the case, start by using the -verbose option with the ./install_stb.sh script so that you get more information of what the installer is doing currently.

If you are still convinced that the installer is hanging, please send a full trace to Oracle. You can do this by:

```
./install_stb.sh -verbose -trace 2>&1 | tee /tmp/trace
```

Do not forget to type **I**(install) to start the actual installation.

### 4.1.6 Package Removal Error in STB Log File (Solaris 11 or Later, Only)

When installing Oracle STB 8.02 on a Solaris 11 domain, the command line output shows:

```
Identified actions for Oracle Explorer Data Collector (8.02):
 - Remove -SUNWexplo
 - Remove -SUNWexplu
 - Remove -SUNWrda
```

The installation log contains messages like:

```
- Failed to remove SUNWexplo
pkgrm: ERROR: unable to change current working directory to
</var/sadm/pkg/SUNWexplo/install>
```

Similar messages can be present for `SUNWexplu`, `SUNWrda`, and `SUNWsneep`. The message is not always present for all packages.

These messages are normal.

The rationale is that it is possible, but not advisable, to install SVR4 packages on Solaris 11 or later. When SVR4 packages were installed, the STB installer will remove the SVR4 package and replace it with the IPS counterpart.

Since the IPS packages have legacy entries, it is not easy to test that the SVR4 packages are already on the system to avoid the message.

Having the legacy entries in the IPS packages, prevents the installation of the SVR4 packages on Solaris 11 or later.

## 4.2 Troubleshooting Explorer Collections

This section addresses known issues and workaround solutions for the Oracle Explorer Data Collector. The following Explorer-specific troubleshooting steps are provided:

- Tracing an Explorer Collection
- Finding Explorer Modules in RDA
- Explorer Error Messages
- ACT Truncated Output
- Running Oracle Explorer Data Collector in the Background
- Explorer Hangs on Zone pkg.ksh (Solaris Only)

### 4.2.1 Tracing an Explorer Collection

When an Explorer collection is running, there are three modes of operation:

- The Explorer engine is running, which sets up the collection environment, starts up the Explorer modules and RDA, and does the post-processing (packaging the collection results and possibly forwarding them).

- One of the native Explorer modules is running (for example, the `sysconfig` module).
- RDA is started, which sets up the RDA collection environment, runs the RDA modules for Explorer, and converts the RDA results to Explorer format.

You can trace each of these modes of operations:

- Engine tracing: set the `EXP_TRACE` environment variable to a non-empty value.
- Native Explorer module tracing (for example, the sysconfig module): set the `EXP_TRACE_<module name in upper case>` environment variable to a non-empty value. For example:

  ```
  EXP_TRACE_SYSCONFIG=1
  ```

  To activate this tracing, it is necessary to also activate the Explorer engine tracing (`EXP_TRACE=1`).
- RDA Explorer tracing: add `Set EXP_TRACE_RDA=1` and export it and/or add `RDA_XPLR_TRACE="1"` as an entry to the `rdainput.txt` file in the Explorer configuration directory.

### 4.2.2 Finding Explorer Modules in RDA

To find out if the collection of an Explorer module is delegated to RDA, issue the following statement:

```
# <RDA location>/rda.sh -X Explorer can
```

This will typically list a series of modules. Typical output would look like:

```
samba ib lp etcextended
```

You can find out the RDA name of the module by inspecting the `mrc.cfg` file. The `mrc.cfg` file is located in the `collect/EXPLORER` subdirectory of the RDA base directory for Oracle Explorer Data Collector 7.3 or higher. Older versions of Explorer keep it in the `modules` subdirectory of the RDA base directory.

> **Note:** For Oracle Explorer Data Collector 7.3 or higher, the collection of all modules is delegated to RDA.

### 4.2.3 Explorer Error Messages

All Explorer error messages are now in the following format:

```
EXPL-nnnnn: ....
```

To get more information on an Explorer error message, run:

```
explorer -X EXPL-nnnnn
```

For example:

```
explorer -X EXPL-00032
```

Output will look like this:

```
EXPL-00032: The temporary directory is not a directory

Cause: The specified location to be used as temporary directory is not a directory
```

```
Action: Remove the specified location or use an alternate location by setting the
EXP_TMP environment variable.
```

### 4.2.4 ACT Truncated Output

Oracle Explorer Data Collector Release 6.6 may truncate the ACT output on certain Solaris 10 systems that has ACT versions between 8 and 8.14 installed. This truncation is done to avoid file system overflow described in ACT bug 6897128.

Work around: Download and install the latest version of ACT into the system.

### 4.2.5 Running Oracle Explorer Data Collector in the Background

If you attempt to run Oracle Explorer Data Collector as a background process in a terminal without -SR, -esc, or -case options, then the tool may wait for you to input (on Explorer type) at the background. In such cases, the tool will not complete its execution and will wait on your input.

Work around: You may note a stopped message on the terminal indicating the suspended process at the background. You will need to bring the suspended process to foreground and then provide the appropriate response to all the Oracle Explorer Data Collector to continue its run.

### 4.2.6 Explorer Hangs on Zone pkg.ksh (Solaris Only)

Launching the Oracle Explorer Data Collector from crontab can sometimes cause the application to hang. This problem is caused by an OS issue, which does not allow crontab directly or indirectly to run:

```
/usr/sbin/zlogin <zone> /usr/bin/pkginfo -l
```

See bug 14674009 in My Oracle Support for more details:

```
https://support.oracle.com
```

The installation of the latest CPU patch and patch 119254-86 solves this issue on Solaris 10.

## 4.3 Troubleshooting SNEEP

This section lists the various errors and issues encountered from using SNEEP:

- SNEEP Performance
- EEPROM `nvramrc` Formatting with Veritas VxVM
- SNEEP Returns Serial Number `unknown`
- Preserving SNEEP Settings After EEPROM Restore or Replace
- Preserving SNEEP Settings After OBP Firmware Update
- Unable to Set `hostname` and `hostid` Tags
- Overriding the Size Limit of the EEPROM `nvramrc` Variable
- SNEEP Reports a `Bad String`

### 4.3.1 SNEEP Performance

On most machines, SNEEP responds immediately; however, with other machines it can take longer.

Some Sun platforms provide the product serial number in the output of the `prtdiag` command. The `prtdiag` command communicates with the Platform Information and Control Library (PICL) daemon, and `picld` may need to communicate with the system controller or ILOM to get current hardware information.

This communication may take a few seconds, with 15 seconds not uncommon.

If SNEEP does not respond for a minute or more, the likely cause is that `picld` is not yet ready, or perhaps never started.

SNEEP may wait for `picld` to respond, retrying `prtdiag` several times before giving up. By default, there will be messages in the system log `/var/adm/messages` indicating these retries.

On Solaris 10, you can find out if `picld` is enabled and functioning by checking the `picl` service. It should look something like this:

```
$ svcs picl
STATE     TIME         FMRI
online    8:10:48      svc:/system/picl:default
```

Prior to Solaris 10, you can verify that the `picld` process is running, but that does not necessarily mean that it is functioning properly:

```
$ ps -ef | egrep 'PI[D]|picl[d]'
UID    PID   PPID C   STIME   TTY   TIME   CMD}}
root   296   1    0   Jan 11  ?     2:26   /usr/lib/picl/picld
```

### 4.3.2 EEPROM `nvramrc` Formatting with Veritas VxVM

If you use Veritas VxVM to manage your storage volumes, and after making a change to the root volume, you may notice that after using SNEEP, the EEPROM `nvramrc` may look strange (for example, everything is flattened onto one long line).

This is an example of the effect of a change made to Veritas VxVM's handling of the EEPROM `nvramrc` variable. This change has been returned to normal with recent VxVM "Rolling Patches" for their Maintenance Packs (MP). Versions of SNEEP before update 1.79 were not prepared for this change, and could lose track of the device alias names for the root volume, which VxVM stores in the eeprom nvramrc. This could even cause a system to fail to boot without manual assistance.

SNEEP update 1.92 and above are not bothered by this change, and can even repair the `nvramrc` variable if you either set a value with SNEEP, or wait for the automatic repair at the next reboot.

**Note:** It is very important to upgrade to at least SNEEP update 1.92.

Usually, the easiest value to set is the serial number, although you could make up any tag and value to set:

```
sneep                   # get the serial number
sneep -s serialnumber   # set it again to correct the nvramrc

or

sneep -t anything -s anyvalue    # make a change to nvramrc
sneep -t anything -s ""          # delete the made-up entry
```

If `sneep -t` reveals any strange SNEEP tags made accidentally by SNEEP releases lower than 1.92 (tags such as `devalias` or `cr`), you can remove them by setting them to an empty value.

Upgrade to SNEEP update 1.92 or newer, then remove them by setting them to a null value.

```
$ sneep -t cr -s ""
$ sneep -t devalias -s ""
```

### 4.3.3 SNEEP Returns Serial Number `unknown`

Depending on your hardware platform, running SNEEP may return a value of `unknown`. Most older Sun platforms have no way to know or report their serial number until after the serial number is manually entered into EEPROM with SNEEP.

Most newer Sun platforms natively provide serial number data through some hardware-based mechanism which can be read by SNEEP, but there are a variety of ways in which this data is provided.

While SNEEP is able to take advantage of many of the mechanisms which provide "hardware" serial number support, at this time, SNEEP does not know how to find serial data on every platform which can provide it in some way.

As part of the ongoing development of SNEEP, support is being added for as many of these methods and platforms as quickly as possible; however, not every mechanism will be supported.

For example, some platforms require a user to interactively log in to a system controller over a public network in order to obtain the serial. Even if SNEEP was capable of this, the configuration required for this is beyond what most users are willing to do.

It is easier for the user to log in manually once to obtain the serial, and then use SNEEP to store that serial into the EEPROM where it can be retrieved easily in the future.

If you are not using the latest SNEEP update, your version may not know how to get the serial number from your machine. A newer update of SNEEP might be able to do this.

There is also the possibility that your machine is one of the platforms for which it is always necessary that the user inform SNEEP of the serial number of the machine using:

```
sneep -s serialnumber
```

This should only need to be done one time, and after that, SNEEP will make it very easy for the user to get the serial number any time that it is needed.

### 4.3.4 Preserving SNEEP Settings After EEPROM Restore or Replace

Your SNEEP settings are preserved if you need to restore your EEPROM to default settings or if you need to replace the EEPROM entirely. SNEEP maintains a backup file `/etc/default/SUNWsneep` in which it keeps a copy of all settings. In case OBP defaults have been restored or if the EEPROM has been replaced without preserving the contents, SNEEP automatically restores the SNEEP EEPROM settings from the backup file when the system is rebooted. Alternatively, the data can be recovered simply by asking for it with SNEEP, and then setting the returned value again with SNEEP.

SNEEP has options designed to make it easy to recover data with very little effort, and under most circumstances it will be able to automatically recover the data at the next system boot. When this is done at system startup, SNEEP will log a message to tell you if the EEPROM is not consistent with the backup and whether or not it was recovered.

See the SNEEP man page for the usage of `-T`, `-d`, and `-P` options.

### 4.3.5 Preserving SNEEP Settings After OBP Firmware Update

Your SNEEP settings are preserved if you need to update your OBP firmware. It is possible to lose the EEPROM settings in an OBP firmware update, but while it once was common, these updates have been very reliable and safe for several years.

If there is a problem, SNEEP will automatically recover the values from the backup.

### 4.3.6 Unable to Set `hostname` and `hostid` Tags

The `hostname` and `hostid` information is always retrieved from the `hostname` and `hostid` commands. To maintain data consistency, SNEEP no longer allows users to set these tags, which can and did create confusing and conflicting situations.

If SNEEP sees that tags exist, they are erased and further requests for the values of these tags are satisfied using the appropriate commands.

### 4.3.7 Overriding the Size Limit of the EEPROM `nvramrc` Variable

By default, SNEEP will prevent you from using too much nvram, but most platforms provide more nvram capacity than SNEEP will allow you to consume. If you are certain that there is more available on your particular platform, you can override SNEEP. If you need a few more bytes than SNEEP allows by default, there is no danger.

However, if you exceed the true maximum capacity of the `nvramrc` variable, some SPARC systems will show no immediate effect, while others will drop into OBP. After that, the system may not boot until the EEPROM defaults are restored:

```
( OBP> set-defaults )
```

A Solaris x86 system may not have an immediate reaction, but may fail to boot later.

---

**Caution:** As these are very serious consequences, it is strongly recommended that you do NOT override the safety limits. If you do so, you are entirely responsible for any system outage or other damages. SNEEP takes care to notify you of this.

---

### 4.3.8 SNEEP Reports a `Bad String`

If SNEEP reports a `Bad String` and is unresponsive, then you probably have an old version of SNEEP and are using a locale or language setting which involves UTF-8. The default `tr` program used in SNEEP did not work in UTF-8 locales. This is no longer a problem after SNEEP 2.5_R1.75.

# A

# Oracle Explorer Data Collector Third-Party License Agreement

This appendix contains licensing information about certain third-party products included with Oracle Explorer Data Collector. Unless otherwise specifically noted, all licenses herein are provided for notice purposes only.

The following sections in this appendix describe the third-party licenses:


- cURL 7.22.0
- OpenSSL 1.0.1p


## A.1 cURL 7.22.0


Copyright (c) 1996 - 2014, Daniel Stenberg.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.


## A.2 OpenSSL 1.0.1p


LICENSE ISSUES

The OpenSSL toolkit stays under a dual license, i.e., both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually, both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL, please contact openssl-core@openssl.org.

OpenSSL License

Copyright (c) 1998-2011 The OpenSSL Project.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.
5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

==========

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Original SSLeay License

Copyright (c) 1995-1998 Eric Young (eay@cryptsoft.com)

All rights reserved.

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are aheared to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed.

If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used.

This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

   "This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)"

   The word 'cryptographic' can be left out if the routines from the library being used are not cryptographic related :-).
4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement:

   "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publicly available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

# Index


## C

CLI installation options, 1-4
collection result file, 2-5
collection result files
  submitting to Oracle, 2-6
configuration directory, 2-4
cron, 2-5

## D

defaults file, 2-4
Diagnotic Assistant (DA), 2-6
download Oracle STB, 1-2

## E

EEPROM
  storing serial number data, 3-3
Explorer configuration directory, 2-4
Explorer Data Collector, 2-1
  configuration, 2-3
  defaults file, 2-4
  modules and groups, 2-5
  nonstandard installation, 2-2
  output directory, 2-4
  security guidelines, 2-7
  standard installation, 2-2
Explorer error messages, 4-5
Explorer software directory, 2-3
Explorer temporary directory, 2-4
Explorer wallet, 2-4
extracting Oracle STB components, 1-5

## F

Fujitsu PRIMEPOWER, 3-6

## H

http/https, 2-7

## I

install
  local zone, 1-4
install Oracle STB, 1-2
IPS installation actions, 1-5

## M

man pages, 2-2
module input files, 2-4

## O

Oracle Explorer Data Collector, 2-1
  configuration, 2-3
  defaults file, 2-4
  nonstandard installation, 2-2
  standard installation, 2-2
Oracle STB
  CLI options, 1-4
  installation, 1-2
  IPS installation, 1-5
Oracle STB components, 1-1
  Oracle Autonomous Crashdump Tool (ACT), 1-2
  Oracle Explorer Data Collector, 1-1
  Oracle Remote Diagnostic Agent (RDA), 1-1
  Oracle Serial Number in EEPROM (SNEEP), 1-1
Oracle Wallet, 2-4

## R

Remote Diagnostic Agent location, 2-3
removing SNEEP tag, 3-4

## S

Secure File Transport (SFT), 2-7
security guidelines
  Explorer Data Collector, 2-7
sharing SNEEP, 3-4
SNEEP, 3-1
  using, 3-3
STB checksum error, 4-1
STB error messages, 4-1
STB installation
  tracing, 4-3
STB_INST_PATH environment variable, 1-4

## T

target directory, 2-5
troubleshooting
  Explorer Data Collector, 4-4
  SNEEP, 4-6
  STB, 4-1

## U

using SNEEP, 3-3